Basic Details

- Full Name: **Amna Gul**
- Email and GitHub Username
    - gul357@gmail.com
    - **Guluna**
- Your first language: **Python**
- Location and Timezone: **Los Angeles, Pacific Time**
- Share links, if any, of your previous work on open source projects: **Sorry, None**
- Convince us that you will be a good fit for this project, by sharing links to your contribution to Sugar Labs: **Sorry, None**

Project Details

- What are you making ?

  I propose to design and build an interactive **AI Art and Comic Creator** tailored for children using the Sugar platform. The activity will use large language models (LLMs) and image generation APIs to empower young learners to create unique, story-driven illustrations and comic panels—even if they lack traditional drawing skills through prompt engineering.

  This activity will allow children to:
  • **Write short story prompts** or select from predefined themes.
  • **Generate AI-assisted images or comic panels** using these prompts.
  • **Arrange them in a visual storytelling format** (single frame or multi-frame comic).
  • **Edit or re-prompt the image** using simple options like "Make it happier," "Add a robot," or "Show a forest at night." or "Create a Studio Ghibli inspired portrait of a young girl" etc.
  • **Save and share** their comic books or art as Sugar Journal entries.

- How will it impact Sugar Labs ?

  This project directly aligns with Sugar Labs' vision of learning through doing. By creating an engaging and user-friendly artistic learning game, we can improve student engagement, combine **creativity + storytelling + AI** in a fun, safe, and educational experience—where every child, regardless of artistic skill, can bring their imagination to life.

- What technologies (programming languages, etc.) will you be using ?
  Programming Language: Python (and Git for version control)
  • **AI Model Interface**:

    - Use an open-source or limited API-based LLM (e.g., GPT-NeoX, Mistral, or local models via HuggingFace)
    - Image generation via Stable Diffusion, DALL·E Mini, or open-source equivalents
- **UI/UX Toolkit**: GTK or Sugar Toolkit (for building interactive UI within Sugar)
- **Storage**: Cloud services like AWS S3 etc for saving art, stories, and comic books
- **Optional Enhancements**: Audio narration using text-to-speech

Timeline:

- Break down the entire projects into chunks and tell us what will you work on each week. As the summer goes on, you and your mentor will adjust your schedule, but it's good to have a plan at the beginning so you have an idea of where you're headed.

Community Bonding Period (May 20 – June 16)
- Get to know the mentors and community
- Study similar Sugar activities
- Finalize game scope and user journey
- Discuss and decide on the tech stack

Coding Phase 1

Week 1-2 (June 17 - June 30)
- Set up development environment and basic UI skeleton
- Implement the game menu, start screen, and activity initialization

Week 3-4 (July 1 - July 14)
- Add addition and subtraction modules with increasing difficulty
- Test and refine the UI based on feedback

Week 5 (July 15 - July 21)
- Implement visual feedback using character animations
- Mid-term testing and bug fixes

First Evaluation Deliverables (July 22)
- Functional prototype with menu, and responsive UI
- Code pushed to GitHub and documentation started

Coding Phase 2

Week 6-7 (July 22 - August 4)
- Add more modules (or test the old ones for robustness)
- Introduce score tracking and rewards system

Week 8-9 (August 5 - August 18)
• Finalize character animations and visual elements
• Add sound effects for feedback and engagement

Week 10 (August 19 - August 25)
• Add polished UI
• Final testing across multiple Sugar environments

Week 11 (August 26 - September 1)
• Clean up codebase, finalize README and wiki
• Submit final version to Sugar Labs repo

Final Evaluation Deliverables (September 2)
• Fully functional game with modules, visual feedback, sound, and scoring
• Complete documentation, code comments, and user guide

- If you will be off-the-grid for a few days, then mention those in the timeline: N/A
- GSoC 2024 has two evaluations, once after every 5 weeks. Highlight the work you plan to complete before each evaluation: Please refer to the timeline section above.
- How many hours will you spend each week on your project ?

  I will dedicate **20-30 hours per week** to the GSoC project, more during critical phases. I'll adjust time if needed based on feedback or challenges.

- How will you report progress between evaluations ?

  Progress will be reported via
  • GitHub commits and PRs
  • Regular check-ins with my mentor via email or chat

- Discuss your post GSoC plans. Will you continue contributing to Sugar Labs after GSOC ends ?

  I do plan to continue contributing to Sugar Labs beyond GSoC—maintaining the project, helping onboard new contributors, and possibly mentoring in the future. This project is more than just a summer commitment; it aligns with my long-term goal of using technology to improve education.